Person Classification Leveraging Convolutional Neural Network for Obstacle Avoidance via Unmanned Aerial Vehicles

Shahmi Junoh[1] and Nabil Aouf[2]

Abstract— **Obstacle avoidance capability for Unmanned Aerial Vehicles (UAVs) remains an active research in order to provide a better sense-and-avoid technology. More severely, in an environment where it contains and involves humans, the capability required is of high reliability and robustness. Prior to avoiding obstacles during mission, having a high performance of obstacle detection is deemed important. We first tackled the detection problem by solving the classification task. In this work, humans were treated as a special type of obstacles in indoor environment by which they may potentially cooperate with UAVs in indoor setting. While existing works have long been focusing on using classical computer vision techniques that suffer from substantial disadvantages with respect to robustness, studies on the use of deep learning approach i.e. Convolutional Neural Network (CNN) to achieve this purpose are still scarce. Using this approach for binary person classification task has revealed improved performance of more than 99% both for True Positive Rate (TPR) and True Negative Rate (TNR), hence, is promising for realizing robust obstacle avoidance.**

I. INTRODUCTION

Robotic systems like drones are susceptible to obstacles during its mission. They may collide with the obstacles and eventually may endanger human beings if the safety aspects are not taken into consideration. One of the important key factors that contributes to successful obstacle avoidance is the accuracy and reliability of the adopted technique of obstacle detection itself. Our approach begins with a classification task of whether or not a person exists at a certain time. In this work, we focus on human beings as the type of obstacle considering humans are inhabitant in indoor environment. This is also in the spirit of applying the first law of Asimov's Laws in that robots should not endanger human by any means. In this work, we consider a single person in this scope – instead of a group of people.

The utilization of deep learning approach in many domains has been shown to be more feasible these days due to (1) Neural Network revisit by Hinton's breakthrough [1], (2) increase in computing capabilities using multi-core CPU and GPUs, and (3) availability of huge dataset collection. It is our belief that resource-constrained systems like UAVs will benefit more and more from the advancement in machine learning domain.

Our person classification approach relevance is twofold: (1) in indoor exploration where humans are considered as a special kind of obstacle that is by any means needed to keep safe, and that requires an accurate and robust detection; and (2) in indoor exploration where humans may cooperate with UAVs to achieve a critical mission like search and rescue, and that requires an accurate and robust detection as well.

II. RELATED WORK

Although classical approaches on computer vision has been popular and dominating for decades in providing classification solution, it is still suffering from robustness problems [2]. In terms of methods that utilize CNN for classification task, our approach is similar to [2] but different in the sense that our work uses visible images while their work uses thermal input data. In [3], they solve classification problem by using CNN for high speed vehicle (~300 fps) which is rather different from our speed requirement. We instead define our speed requirement to be somewhat in decent time (~5 fps) due to the fact that indoor exploration does not require a fast vehicle motion.

In the context of obstacle avoidance there are work like in [4], that attempt to learn high-level steering command required like turn right, turn left, or go straight when confronting an image containing obstacle. We, however, approach the problem differently i.e. by first classifying the obstacle and then detecting and finally tracking it. That leads to the localization of the obstacle and to the prediction of a collision-free trajectory.

As opposed to people detection problem that has been tackled by using camera mounted on a non-moving system, by using such a UAV system it normally imposes input images with different types of challenges like blurry, highly rotated or even interlaced. Several attempts that show how deep learning approach has been applied on such a non-moving system are like in [5]–[8].

III. PERSON CLASSIFICATION BY CNN APPROACH

We explored deep learning frameworks like Deeplearning4j [14], Caffe [9], MxNet [10], and TensorFlow [11] with the goal of choosing a computationally efficient one. We focused first on time efficiency and development effort complexity. Our investigation has led to Caffe as a first choice.

*This work was supported by Centre of Electronic Warfare, Information and Cyber, Cranfield University, Defence Academy of the United Kingdom, UK and Universiti Teknikal Malaysia Melaka, Malaysia

[1]Shahmi Junoh is with Centre of Electronic Warfare, Information and Cyber, Cranfield University, Defence Academy of the United Kingdom, Shrivenham SN6 8LA, UK (s.junoh@cranfield.ac.uk) and with Faculty of Engineering Technology, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

[2]Nabil Aouf is with Centre of Electronic Warfare, Information and Cyber, Cranfield University, Defence Academy of the United Kingdom, Shrivenham SN6 8LA, UK (n.aouf@cranfield.ac.uk)

A. Camera

Two different cameras were used. The reasons behind this were firstly, to see how it would perform on a less complicated setup for fast prototyping our approach i.e. using phone's camera (as opposed to using drone's camera directly) and secondly, to observe how transfer learning was gained. Camera from Samsung Galaxy Note 3 was used to gather our dataset to build our first model i.e. PersonNet. The sensor used in that phone camera is CMOS with 13 MP. To build our second model i.e. PersonNetUAV, front-facing camera from Parrot AR.Drone[1] platform was utilized. The sensor is CMOS with 1280x720 pixels which has almost 14 times lower resolution than that of Samsung Galaxy Note 3 camera.

B. Dataset

For PersonNet model, three sets of dataset were generated and grouped: 1) training set, 2) validation set, and 3) test set as shown in Fig. 1. Examples contain 1000 images, both for positive and negative example making up 2000 images in training set in total. Likewise, in validation set, it has 1000 images, both for positive and negative example making up 2000 images in sum. Along with the dataset, labels were prepared and assigned to correspond to a person and the other way around accordingly.

Likewise, the above process was repeated for PersonNetUAV model. Three sets of dataset were generated and grouped: 1) training set, 2) validation set, and 3) test set. Fig. 2 illustrates some of training examples used to train PersonNetUAV classifier. Examples contain 697 images, both for positive and negative examples making up 1397 images in total in training set. Similarly, in validation set, it has 697 images, both for positive and negative example making up 1397 images in sum. Labels were also prepared and assigned accordingly.

For both classifiers, from the generated dataset, a database is then created so as to have an input compatible and efficient with the network (Details of network will come later). A database is composed of image and label pair so as to provide a guide to the CNN system during training phase and a verification on how well the most updated learned model has been doing during validation phase. Then, a mean over the training images was computed.

It is also noted that prior to feeding the input to the network to let it train, we normalized input in the hope to have intensity values in the interval around [-128, 128] and have mean value around 0. It will, in principle, help achieve a shorter convergence time. We obtained that by performing mean subtraction per color channel on every input image per pixel basis such that for each color channel of R, G and B

$$(R, G, B) := (R - R_\mu, G - G_\mu, B - B_\mu) \qquad (1)$$

where R_μ, G_μ and B_μ are mean values of respective channels.

[1]https://www.parrot.com/uk/drones/parrot-ardrone-20-elite-edition#parrot-ardrone-20-elite-edition

Furthermore, scaling on input images of 227×227 pixels was also applied. A center crop 227×227 pixels on input images was performed before inputting to the CNN network. Random horizontal flip was also applied during training to increase the transitional invariance robustness. Although other image preprocessing techniques like histogram equalization may help for further processing, we skipped that as we believed that we would not gain much by doing that.

C. Architecture

While the inherent problem of classical Artificial Neural Network (ANN) possesses is too tedious to fine tune parameters, CNN has a good solution for that i.e. by the notion of transfer learning. Therefore, we decided to build on top of an existing reference model offered by Caffe framework [9] called CaffeNet. CaffeNet is a modified version of AlexNet [12] and they are different only in the sense that CaffeNet has been trained without data augmentation and pooling and normalization layers are swapped. While the CaffeNet has number of outputs of 1000 in the final fully-connected layer, we changed ours to 2 to suit our binary classification task. Fig. 3 depicts our resulting architecture.

One of the benefits by using AlexNet as a baseline architecture is in the neuron's output modeling. The employment of nonlinearity element of Rectified Linear Units (ReLUs), which is $f(x) = max(0, x)$, was reported to have performed six times faster in the training error rate than the previously known $f(x) = tanh(x)$ where f is an output function of input x [12].

D. Training Details

Stochastic gradient descent was utilized during training with batch size of 256, momentum of 0.9 and weight decay of 0.0005. We found that the convergence is obtained at the iteration of 80 for PersonNet and 40 for PersonNetUAV. We used only CPU mode for training – instead of GPU, and yet successful trained our network without suffering from vanishing gradient problem. Fig. 4 shows the training performance until the training finally converged at the 80th iteration for PersonNet classifier.

E. Classifiers

Two resultant classifiers were generated called PersonNet and PersonNetUAV. PersonNetUAV was trained on top of PersonNet. Both are different in that: the dataset used to train the network that results in PersonNet classifier uses a camera phone which has far higher resolution compared to the dataset used to build PersonNetUAV model which uses on-board Parrot AR.Drone's camera, the dataset used to build PersonNet classifier is calibrated while the dataset used to build PersonNetUAV is uncalibrated, the dataset used to build PersonNet classifier used about 30% of training data more compared to the dataset used to build PersonNetUAV model and both were trained in different indoor environments.

Fig. 1: First three columns from left are some of the training images in the training set whereby next three are of validation set images. Images at the seventh column are test images (top image being for testing on true positive and bottom on true negative) that outputs the score in percentage during prediction. Images at the top are positive instance while the ones at the bottom are negative.

Fig. 2: Images at the top are examples of positive training data while ones at the bottom are negative training set.

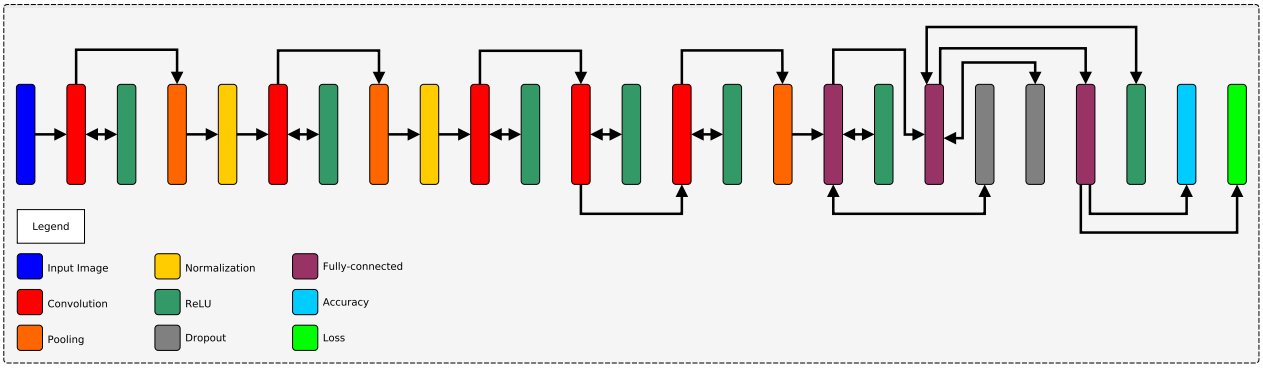


Fig. 3: Our CNN architecture that was built on top of CaffeNet.

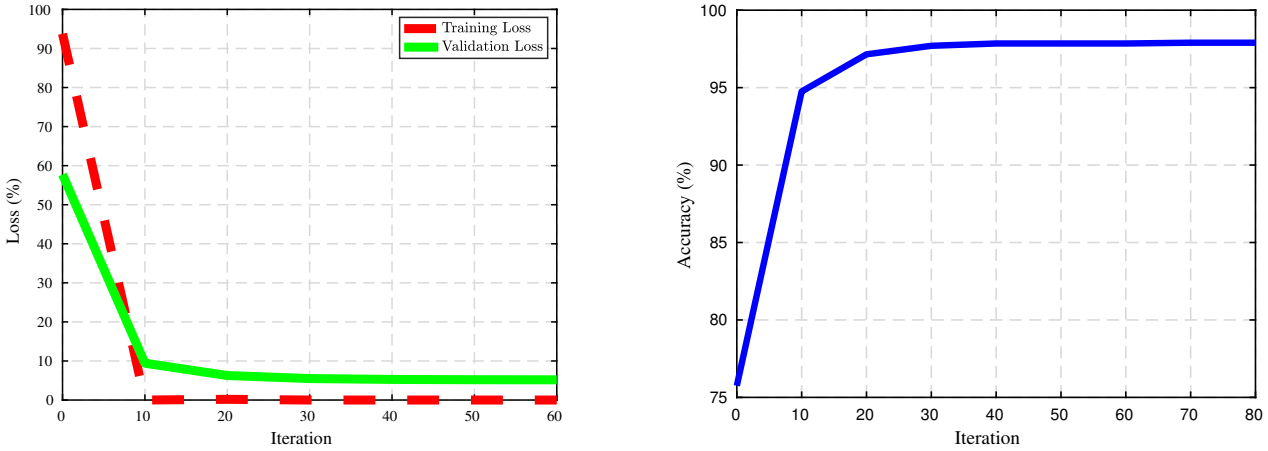
IV. EVALUATION

A. *Quantitative Result*

Both PersonNet and PersonNetUAV classifiers were tested on two sets of images. The first set contains 1347 images by which it is dedicated for test on positives while the same amount of images are for test on negatives. Images whose test is on positives are images that contain a person which is not seen before by the trained classifier. Test on negatives involves test on images of the indoor corridor environment that the classifier has never seen before.

Table I shows the performance of our PersonNet classifier quantitatively. The FN is found to be very low i.e. 11 relative to the total images (1347 images). Table II summarizes the obtained results. The classifier achieved a very good performance especially in the TPR and TNR and it was shown to be robust.

More interestingly, PersonNetUAV classifier outperformed the performance of PersonNet on the same test dataset. Table III and Table IV show a comparative overview of both



(a) Loss comparison (b) Model accuracy

Fig. 4: Comparison of loss between training and validation phase is in (a) while in (b) is the model accuracy upon achieving its convergence. For the sake of illustrating the training performance, we only show for the case of PersonNet classifier, however for PersonNetUAV it behaves similarly but with shorter convergence time.

TABLE I: CONFUSION MATRIX OF PERSONNET

		Prediction	
		Person	Negative
Ground truth	Person	1336	11
	Negative	0	1347

TABLE II: THE OBTAINED PERFORMANCE OF PERSONNET

Metric	Result (%)
True Positive Rate (TPR)	99.18
True Negative Rate (TNR)	100
Positive Predictive Value (PPV)	100
Negative Predictive Value (NPV)	99.19
False Positive Rate (FPR)	0
False Discovery Rate (FDR)	0
False Negative Rate (FNR)	0.82

TABLE III: CONFUSION MATRIX OF PERSONNETUAV

		Prediction	
		Person	Negative
Ground truth	Person	1342	5
	Negative	3	1344

TABLE IV: THE OBTAINED PERFORMANCE OF PERSONNETUAV

Metric	Result (%)
True Positive Rate (TPR)	99.63
True Negative Rate (TNR)	99.78
Positive Predictive Value (PPV)	99.78
Negative Predictive Value (NPV)	99.63
False Positive Rate (FPR)	0.22
False Discovery Rate (FDR)	0.22
False Negative Rate (FNR)	0.37

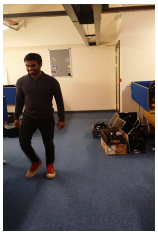
performances. In comparison to PersonNet, PersonNetUAV classifier reduced the FN from 11 to 5.

Since the nature of building exploration whereby humans are more likely to surround or operators may co-operate with UAVs, hence it is an important requirement to have a very high TPR. While both TPR and TNR have to be very high (preferably, close to 100%) for real deployment, having a very high TPR is considered more severe to be fulfilled. This is to avoid UAVs hit humans accidentally in a more systematic way [2].

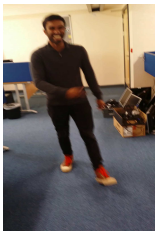
B. Qualitative Result

While the quantitative result that we have just presented before was convincing, here we intend to present some key observations on how the classifier performed on some challenging scenarios qualitatively. Fig. 5 illustrates those test images that are considered challenging and yet the classifier was able to perform very well. Despite the person's image is off-centered like in Fig. 5(a), the PersonNet classifier was able to work very well whereby it scored 99.99% for that. With blurry person's image like in 5(b) and even with jumping like in Fig. 5(f), the classifier could perform well. Even with other challenging pose shown in Fig. 5, the PersonNet classifier demonstrated to work well.

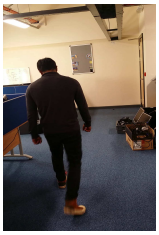
Nevertheless, there are cases whereby it had false negative as shown in the last column of that Fig. 5. Our assumption is that it could be the case somehow if the person's image contained in the image is too blurry like in Fig. 5(d), then the classifier does not perform as intended. Looking from indoor exploration using UAVs point of view, there are several ways that we can go about it. One of the possibilities is to do naive filtering (sampling) i.e. not only to depend on classification score that is based on only one image, but rather a few, e.g. sampling over 3 consecutive images. However, in order to make it more accurate, the use of advanced filtering schemes like Kalman Filter is useful.



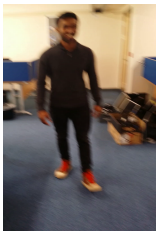
(a) Off-centered with 99.99% score



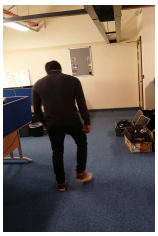
(b) Blurry with 84.52% score



(c) From back with 88.81% score

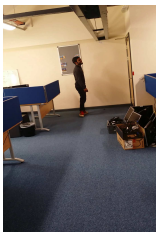


(d) False negative



(e) Head bending with 85.41% score

(f) Jumping with 86.16% score



(g) From side with 100% score

(h) False negative

Fig. 5: Some different challenging scenarios tested with good result scored by the classifier. The last column are those with false negative.

TABLE V: CLASSIFICATION ACCURACY COMPARISON OF PERSONNET AND PERSONNETUAV ON THE SAME CHALLENGING TEST SET

	Accuracy (%)	
Scenario	PersonNet	PersonNetUAV
(a)	99.99	98.56
(b)	84.52	94.99
(c)	88.81	98.83
(d)	FN	70.57
(e)	85.41	99.02
(f)	86.16	87.45
(g)	100	99.93
(h)	FN	FN

We also observed that apart from those challenging cases, the PersonNet classifier was shown to be of high accuracy and robustness on positives. For evaluation on negatives (to evaluate the TNR), our dataset are similar to what has been shown in Fig. 1 at the bottom row. At the moment, the TNR is 100% for which we plan to test on negatives on much larger and diverse dataset.

Likewise, the PersonNetUAV classifier also showed a similar performance on that challenging images and even better for some of those. Table V summarizes the performance of both methods.

C. Discussion

We can conclude that the transfer learning has proven to be useful in our application. Despite the Caffe reference model has not been originally trained with dataset containing person, it still performed well. This is primarily because low level features embedded in lower layers of CNN network share many common features. That is the reason why we had trained it with our own dataset for our classification problem so as to help higher layers learn more relevant high level features with respect to our target class.

The research has led to several interesting observations concerning the performance of PersonNetUAV classifier which was derived from PersonNet. With PersonNetUAV, it was able to work with challenging images which failed with PersonNet that incurred false negative. More specifically, 11 of FN was reduced to 5 and that gives a better confidence and robustness in the classification capability of the classifier. Despite the PersonNetUAV was trained in different environment, with different camera and the dynamic coupled with the hovering UAV, the classifier is yet able to perform well.

Furthermore, although uncalibrated images were used to train PersonNetUAV, it is still able to work well and in fact even better than PersonNet classifier in some scenarios (see Table V). The *barrel* effect of radial distortion in uncalibrated images can be seen in the input images as shown in Fig. 2.

Comparing to similar approach in the recent literature, e.g. [2], whose their TPR and TNR are both 81% and 92% respectively for person classification task, our classifiers perform much better i.e. higher than 99% both for TPR and TNR accordingly.

With respect to the demand of computational price, we believe that for such an indoor exploration, the flight operational speed – computational price is positively correlated – does not have to be very fast but be in decent speed. Otherwise, other information like mapping information and related may be missing while accomplishing the mission. In [13], they prove the feasibility of using CNN on UAVs computationally although their use case is different i.e. learning controller strategy while searching a target.

V. CONCLUSION

We have presented the task of person detection which has long been dominated by classical computer vision approaches by using deep learning approach. The importance of this work lies in the person classification task for the use of UAVs in indoor environment, specifically for human obstacle avoidance. This person classification solution does not only aim to help toward collision-free navigation but also to better enable situation awareness where human and UAVs may co-operate to accomplish complex mission like search and rescue. We showed that this approach resulted in very high performance of more than 99% both for True Positive Rate (TPR) and True Negative Rate (TNR) for binary person classification task. In addition to that, we obtained an interesting result whereby despite uncalibrated images have been used to train our network, its performance does not differ so much compared to that has been trained with calibrated ones.

This work has been dedicated for single person classification and may be extended for multiple persons in the future. Furthermore, challenges like occlusion, classification of people with scarf and toddlers can be tackled later.

While the obtained performance is convincing, we plan to test our trained classifier on much larger and diverse datasets. Although the datasets have been taken from a real UAV and used offline, we are eager to see how it behaves using the generated classifier online.

ACKNOWLEDGMENT

This work was supported by Centre of Electronic Warfare, Information and Cyber, Cranfield University, Defence Academy of the United Kingdom, UK and Universiti Teknikal Malaysia Melaka, Malaysia. We would like to thank our colleagues Duerte Rondao and Raymond Vincent for their help with the dataset collection.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks." [Online]. Available: https://www.nvidia.cn/content/tesla/pdf/machine-learning/imagenet-classification-with-deep-convolutional-nn.pdf

[2] D. C. D. Oliveira and M. Aur, "Towards Real-Time People Recognition on Aerial Imagery using Convolutional Neural Networks," 2016.

[3] M. Mancini, G. Costante, P. Valigi, and T. A. Ciarfuglia, "Fast Robust Monocular Depth Estimation for Obstacle Detection with Fully Convolutional Networks," pp. 4296–4303, 2016.

[4] K. Sullivan and W. Lawson, "Deep Obstacle Avoidance âĹŮ," 2013.

[5] R. Stewart, M. Andriluka, and A. Y. Ng, "End-to-end people detection in crowded scenes," in *CVPR*. IEEE Computer Society, 2016, pp. 2325–2333. [Online]. Available: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7776647

[6] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *ICCV*. IEEE Computer Society, 2015, pp. 1904–1912. [Online]. Available: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7407725

[7] ——, "Pedestrian detection aided by deep learning semantic tasks," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 5079–5087.

[8] D. Tomè, F. Monti, L. Baroffio, L. Bondi, M. Tagliasacchi, and S. Tubaro, "Deep convolutional neural networks for pedestrian detection," *Image Commun.*, vol. 47, no. C, pp. 482–489, Sep. 2016. [Online]. Available: https://doi.org/10.1016/j.image.2016.05.007

[9] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[10] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang, "MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems," *CoRR*, vol. abs/1512.01274, 2015. [Online]. Available: http://arxiv.org/abs/1512.01274

[11] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," May 31 2016, comment: 18 pages, 9 figures; v2 has a spelling correction in the metadata. [Online]. Available: http://arxiv.org/abs/1605.08695

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[13] D. K. Kim and T. Chen, "Deep Neural Network for Real-Time Autonomous Indoor Navigation."

[14] Deeplearning4j Development Team. Deeplearning4j: Open-source distributed deep learning for the JVM, Apache Software Foundation License 2.0. http://deeplearning4j.org